7. Integration and Management of Transactional Government Services

7.1. Service Oriented Transactional Government Services

One of the important objectives of the State Portals would be to provide access to services offered by government departments and organizations such as “applying for birth certificate”, “applying for electricity connection” etc. State government departments are the owners of these processes. Level of automation of these processes will vary from state to state and also departments within a state. Further departments which have adopted computerization and automated their processes will be using different set of architecture styles, tools and technologies. Under this scenario State Portals need to use different combination of integration approaches for accessing departmental applications and processes.

The State Portal would use service oriented architecture to access departmental applications. Departmental applications would expose their functionality as service. Services would interface with departmental applications, providing access to department’s services and processes. Using open standards and service oriented architecture will help in realizing the principles of interoperability, reusability, extensibility, loose coupling, technology independence, vendor independence etc.

7.1.1. Web Service Compliant Services

Services would be developed using web services technology or SSDG technology. As far as E-Form application under this project is concerned, it has to use the SSDG as the middleware. The application architecture will be as per the appendix C.

A Web service is a software standard designed to support interoperable machine-to-machine interaction over network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards. web based services would be deployed on a web service compliant “common service communication infrastructure” using a federated deployment model, where a service cluster may exist at different levels such as national, state, department, district, block etc. Key advantage of using web services will be adopting open standards to:

a. Describe services (WSDL)
b. Publish services to registry and repository (UDDI)
c. Lookup services into registry and repository (UDDI)
d. Invoke services and (SOAP)
e. Describe business process by means of orchestrating services (BPEL)

7.1.2. SSDG Compliant Services

Service may be developed based on SSDG technology. SSDG based services would be deployed on SSDG compliant “common service communication infrastructure”. As far as E-Form application under this project is concerned, it has to use the SSDG as the middleware. The application architecture will be as per the Appendix C.

7.2. Integration with Departmental Applications

NSDG Specifications: http://www.nsdg.gov.in/administration/technicaldetails.jsp

7.2.1. General Interface for Transaction Government Services

State Portals would work as a single source of information for all government content and provide front end or start point for all the government services provided by the various state government departments. It is required that functionality of departmental applications and services provided by the departments are made accessible on State Portal as services using web services technologies. Web service based implementation of transactional government services will make them reusable, easy to integrate, interoperable and easily accessible. Any government website, portal or application would be able to easily consume these transactional government services using standard protocols. Below figure shows integration architecture for communication between State Portal and departmental applications.

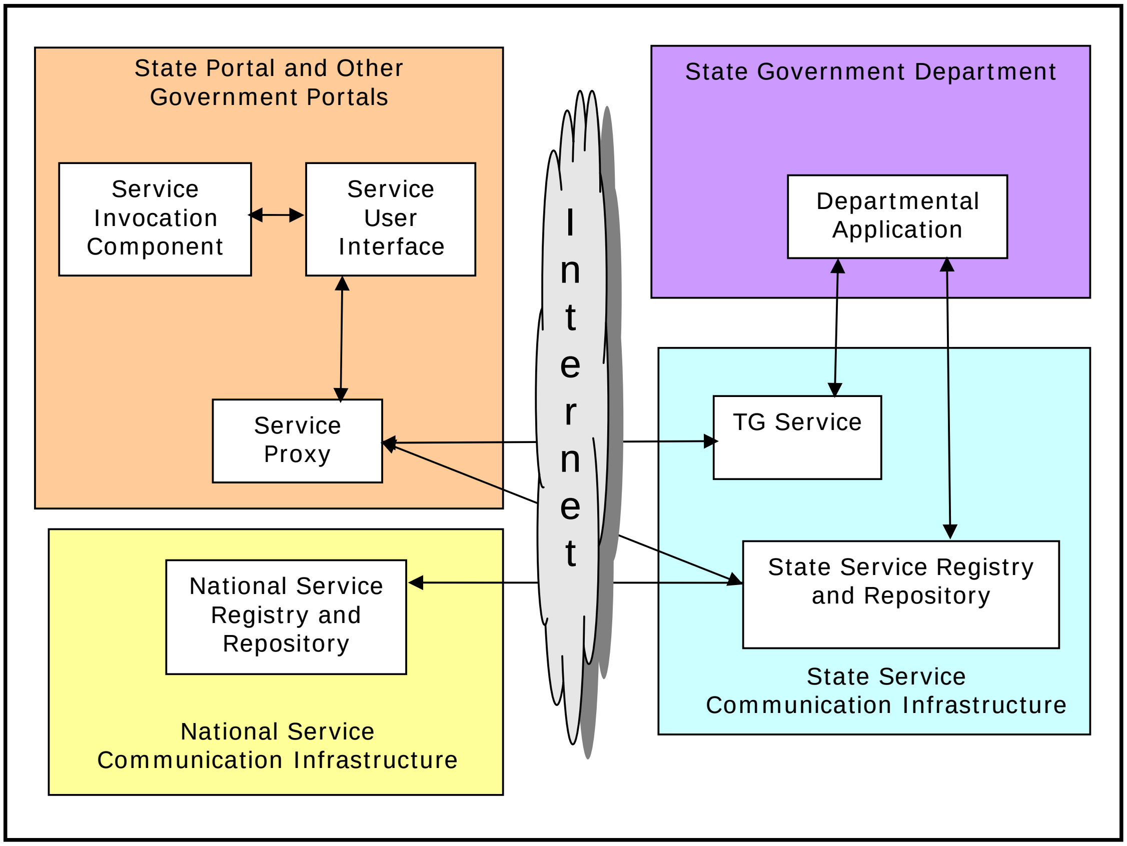
State Portal and Other Government Portals
Service Invocation Component
Service User Interface
Service Proxy
National Service Registry and Repository
National Service Communication Infrastructure
Internet
State Government Department
Departmental Application
TG Service
State Service Registry and Repository
State Service Communication Infrastructure

Figure 10. Integration with Departmental Applications

Transactional government services should be hosted on state level common "service communication infrastructure". Similarly National level services such as services related to railways, parliament, passport etc. would be hosted on national level common "service communication infrastructure". This makes all state and national level services accessible to all government applications and portals. This model would help in achieving high level of reuse of services at intra state, inter state and national levels.

7.2.2. Service Invocation Component

Service invocation component would be a technology entity capable of displaying dynamically created content such as JSP, ASP.Net, Servlet, PHP etc.

7.2.3. Service User Interface

This represents the web based interface for transactional government services. This would be implemented using technologies like JSP, Servlet, PHP, ASP etc. It's exactly functionality would vary from service to service. It should confirm to defined user interface guidelines.

7.2.4. Service Proxy

Service proxy represents transactional government service on service consumer side (such as State Portal) encapsulating all technology related details of service such as getting service URL, handling exceptions etc.

7.2.5. Departmental Applications

This represents any departmental application of state government departments.

7.2.6. State Service Registry and Repository

This represents state level directory of transactional government services. This should support open standards UDDI and ebXML. Service repository should provide web based interface.

7.2.7. National Service Registry and Repository

This represents national level directory of transactional government services. Content from state level service registries and repositories will be replicated into National service registry and repository. Service registry should support open standards UDDI , ebXML. Service repository should provide web based interface.

7.2.8. Transactional Government Service (TG Service)

Transactional government services should be implemented as web service. It will expose functionality of departmental applications to other applications and portals. Transactional government service should provide following common operations, however operations parameters may vary from service to service. e-Form applications in this project shall be governed by the SSDG implementation.

7.3. Implementation Scenarios

Implementation approach for transactional government services depends on architecture and technologies used for developing departmental applications. This section explains some of the integration approaches and scenarios under which they should be adopted for developing transactional government services.

Current State	Suggested Integration Approach
Departments are yet to start using computers	Web-form integration
Departments are using only email and office tools; yet to automation of business processes	Web-form integration with email notification
Departmental applications are based on client server architecture	Data level integration
Departmental applications are based on three tier architecture but not web based	Application level integration reusing DLLs, libraries etc.
Departmental applications are based on N-tier architecture with web based user interface	Application level integration reusing EJBs, .Net components, COM+ components etc.
Departmental applications are service based	Service level integration

7.4. Web Form Integration

An Web Form is a computer version of a paper form, which can be filled online, submitted and print out of filled form can be taken for manual submission (if required). Some of the benefits of Web Forms are eliminating the cost of printing, storing, and distributing pre-printed forms, reducing consumption of paper, eliminating the need of disposable of obsolete forms. Web Forms can prevent most of the data quality related errors, provide required help and guidance for correctly filling forms, automatically perform format conversions, calculate, look up, and validate information for the user.

Further using Web Forms along with digital signatures, form submission, routing to required section, department or authority can be done electronically, which means further savings in terms of cost and cycle time of service. Departments where already computerization is implemented Web Forms can eliminate the cost of re-keying data and the associated errors.

It is possible that some of the departments are yet to start using computers and all their processes are still paper based. In such a scenario, State Portals should provide electronic submission of requests or application forms on the portal. This form of integration can be termed as Web Form Integration. This should be the first level of service integration. Following figure shows the web form integration approach.

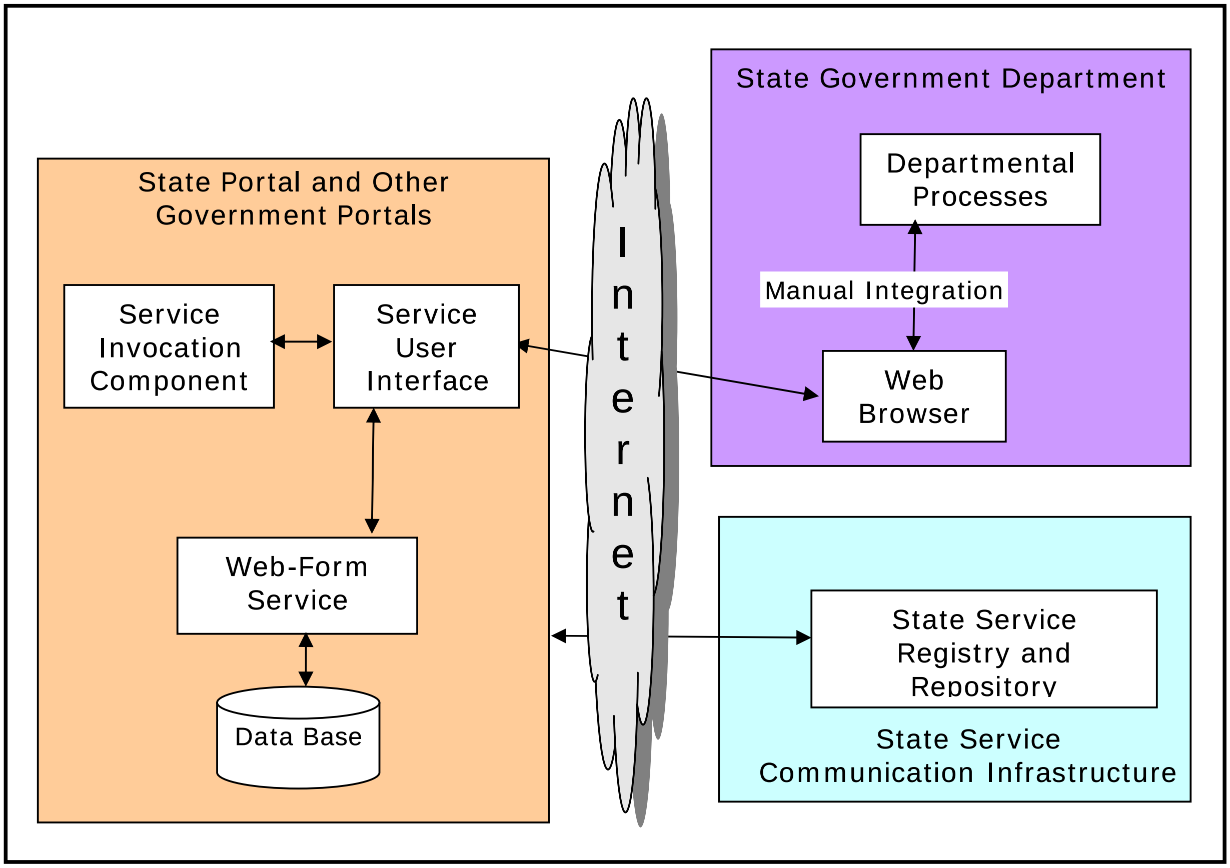
State Government Department
Departmental Processes
Manual Integration
Web Browser
State Portal and Other Government Portals
Service Invocation Component
Service User Interface
Web-Form Service
Data Base
Internet
State Service Registry and Repository
State Service Communication Infrastructure

Figure 11. Web Form Integration

7.4.1. Web-Form Service

In this case IT ownership of entire service would be with State Portal. Since department processes will be manual, manual integration will be done where in department will be provided web based access to the service using a web browser. Further email based notification should be implemented when departments have access to email and office automation tools.

7.4.2. Usage Scenarios

Web form services should provide following functionality:

a. Fill and submit form.
b. View and print submitted forms/requests.
c. Generate a unique tracking number for every submitted form.
d. Cancel submitted requests/forms.
e. Display status or submitted requests/forms.
f. Update status, to be used by respective department.
g. Delete and archive submitted requests/forms, to be used by administrators.

Following figure depicts various usage scenarios of services for which web form integration approach would be adopted.

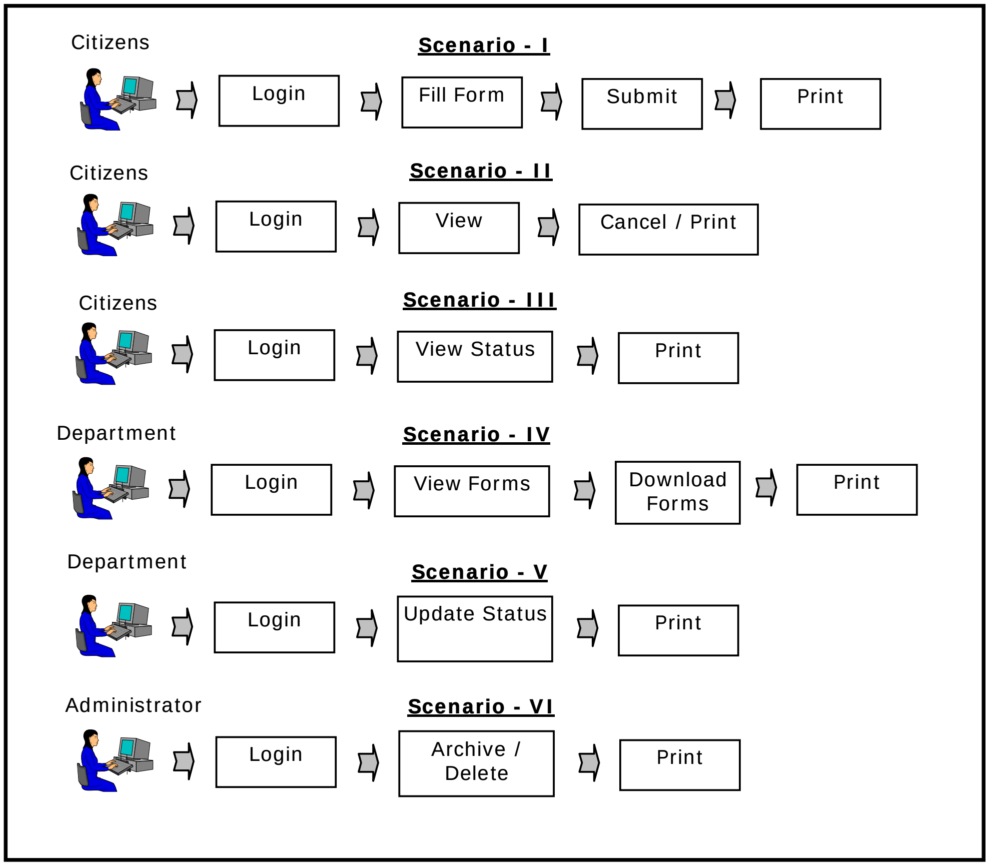

Figure 12. Illustrative Web Form Service Usage Scenarios

7.5. Data Level Integration

In this integration approach State Portal (external application) communicates with an existing application's data or data tier, through a newly developed software entity such a service. Data level integration results in redevelopment or reengineering of business logic and subsequently maintaining two versions of code and additional effort for keeping them in synch. Design has to make sure that transactional integrity is maintained while old and external application accesses data simultaneously. A major advantage of this approach is that it does not require any modifications of the existing applications. Furthermore, the approach relies on inexpensive and established technology, i.e. database oriented middleware such as ODBC and JDBC.

Some of the state government departments have adopted computerization long time back, when client server architecture and technologies were widely in use. These departments will have applications developed on client server architecture using technologies such as Visual basic, Visual, FoxPro, PowerBuilder etc. These applications by design combine business logic and presentation or user interface tiers, whereas data tier is well separated. In general these applications use some older versions of relational database management products such as Oracle, SQL server, Informix, Ingress, Gupta SQL, Sybase etc.

Integrating client server architecture based legacy applications with State Portals would pose few challenges due to reasons like

a. Non availability of technical documentation
b. Poor quality of source code

c. Absence of people, who are familiar with architecture/design and source code
d. In few cases, even source code may not be available etc.

State portal would consider using data level integration under following scenario

i. Departmental application using client server architecture.
ii. Application or business logic not well separated.
iii. Application code is not modular and difficult to maintain.
iv. Data schema can be decoded
v. Database system in user supports transactions and multiple concurrent connections.

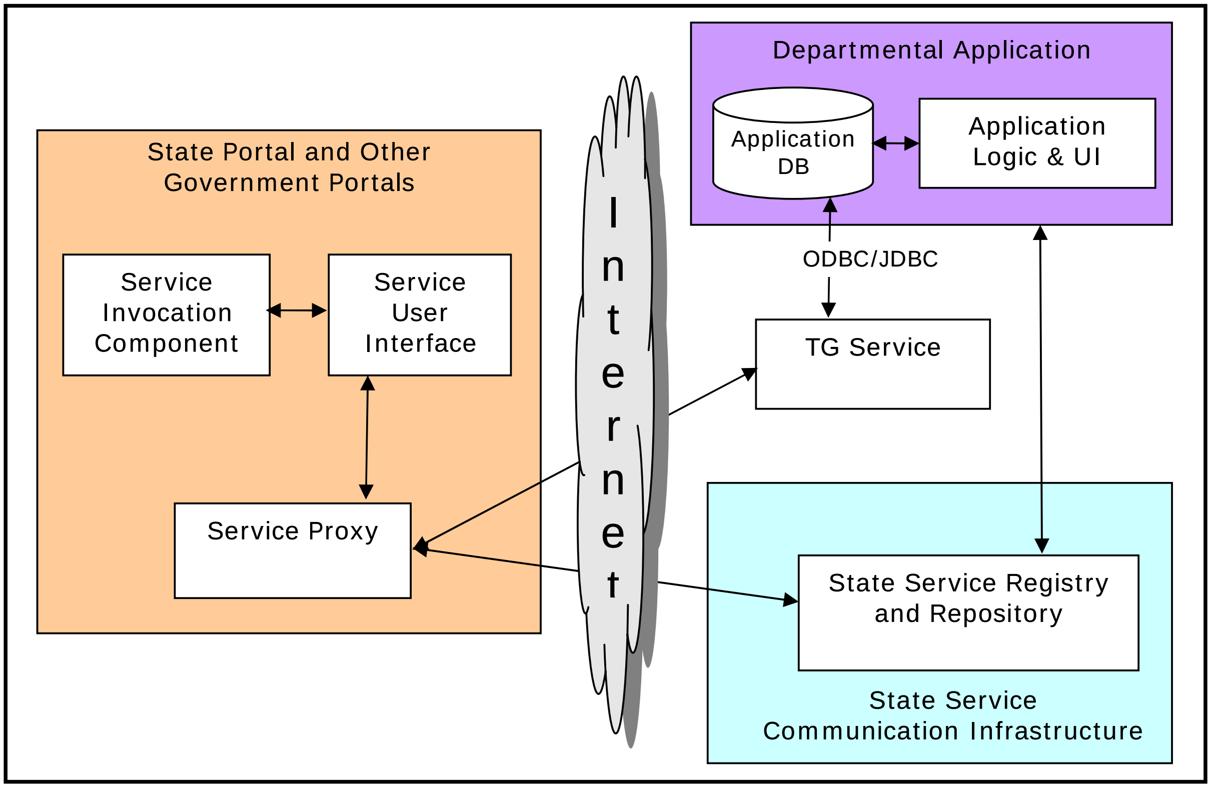
Departmental Application
Application DB
Application Logic & UI
ODBC/JDBC
TG Service
State Portal and Other Government Portals
Service Invocation Component
Service User Interface
Service Proxy
Internet
State Service Registry and Repository
State Service Communication Infrastructure

Figure 13. Data Level Integration

Transactional government service (Web service) will integrate with existing application at data level. It will access application data using standard protocol such as ODBC and JDBC.

7.5.1. Application Logic and UI

In this scenario departmental application will be based on client server architecture. It will be using technologies like VC++, VB, Power Builder, FoxPro, Delphi, Oracle Forms, PHP etc. Application logic and user interface will be tightly coupled.

7.5.2. Application DB

Technologies used for application DB will be Oracle, SQL server, FoxBase, Sybase, Ingress, Informix etc.

7.6. Application Level Integration

In this integration approach State Portal (external application) communicates with an existing application's components such as API, business logic components or business logic tier, through a newly developed software entity such as service. Application level integration reuses existing code and may extend it for the purpose of data conversion, protocol conversion etc.

7.6.1. Integration using Application Logic Components

State government department's applications, which are built on three tier architecture or applications whose business logic code is well separated from presentation and data tier, become good candidates for adopting application level integration. From the implementation or technology point, business logic may exist in form of DLLs, functions, CORBA, COM, .Net of J2EE components etc. In this scenario transactional government service will be a wrapper on top of the existing code interfacing with external applications using standard protocols.

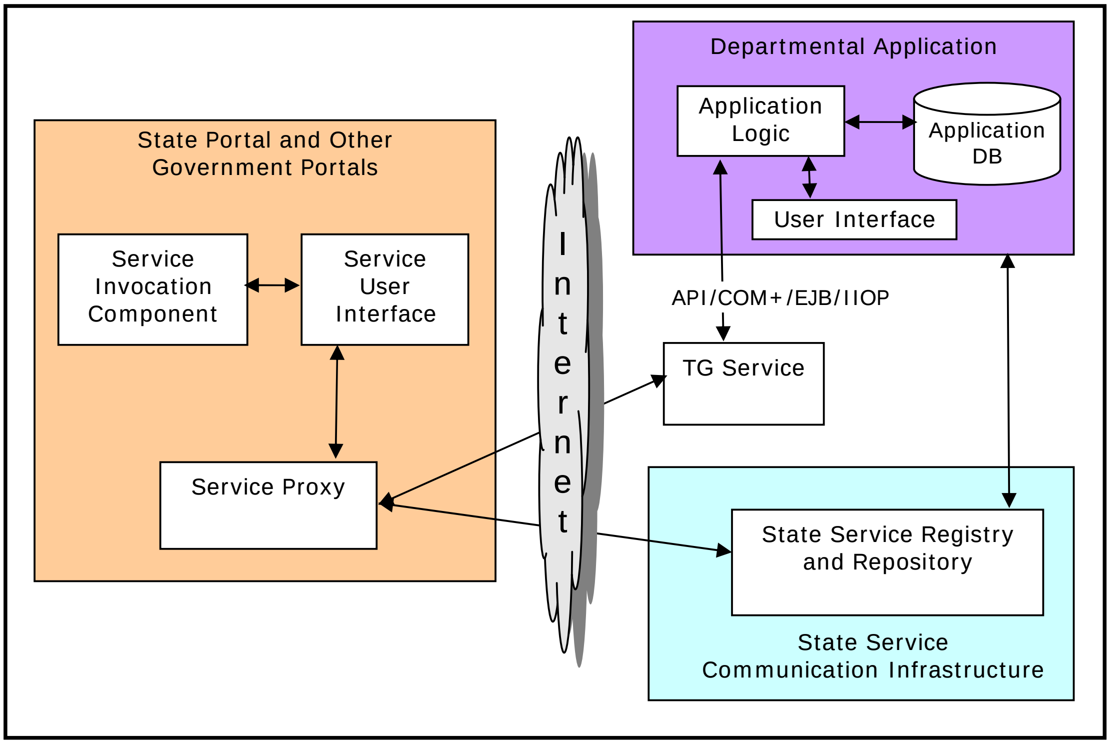
Departmental Application
Application Logic
Application DB
User Interface
State Portal and Other Government Portals
Service Invocation Component
Service User Interface
Service Proxy
Internet
API/COM+/EJB/IIOP
TG Service
State Service Registry and Repository
State Service Communication Infrastructure

Figure 14. Application Level Integration

7.6.2. Application Logic

In this scenario departmental application will be based on three tier architecture. It will be using technologies like COM, COM+, .Net, EJB, CORBA, shared DLLs, libraries etc. for implementing logic. Application logic and user interface will be clearly separated using a well defined interface. Business logic will reside in the form of components or well defined functions or procedures packaged as static or dynamically linked libraries.

7.6.3. User Interface

User interface will be either web based on rich GUI.

7.6.4. Application DB

Technologies used for application DB will be Oracle, SQL server, DB2 etc.

7.7. Integration with PHP Application

Following figure shows integration with PHP based applications.

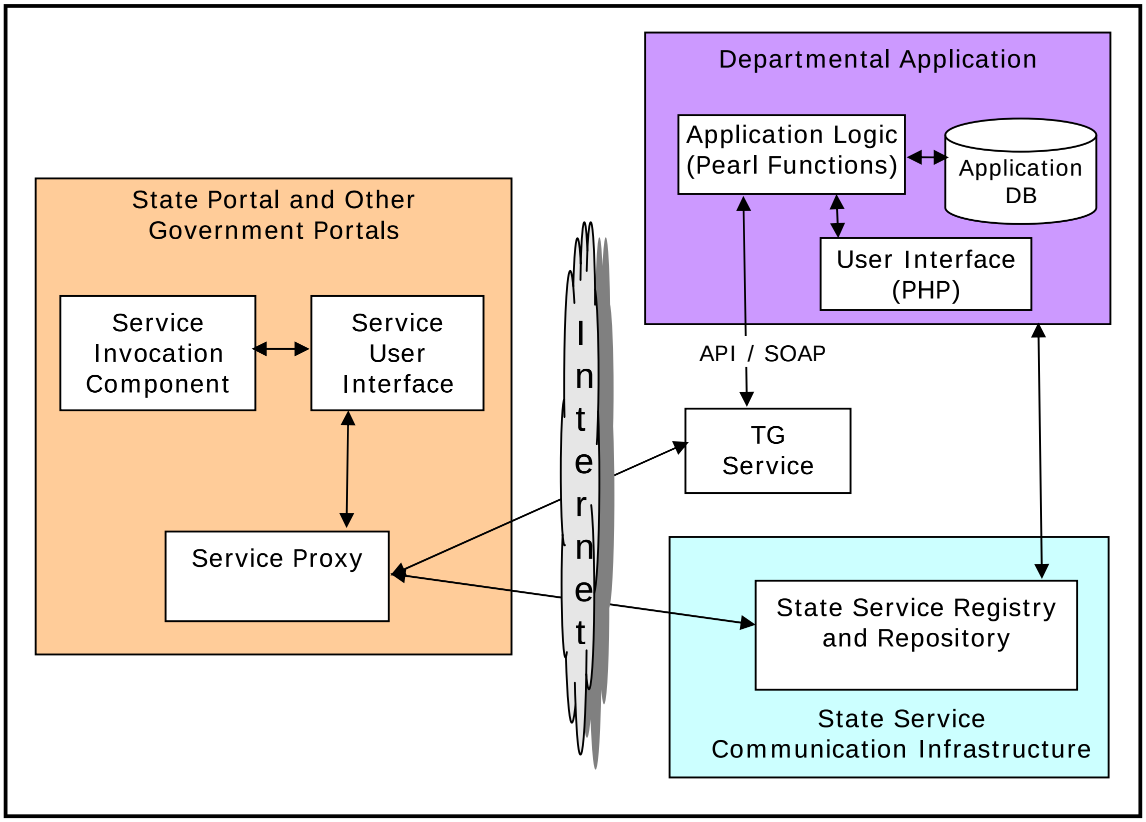
Departmental Application
Application Logic (Pearl Functions)
Application DB
User Interface (PHP)
State Portal and Other Government Portals
Service Invocation Component
Service User Interface
Service Proxy
Internet
API / SOAP
TG Service
State Service Registry and Repository
State Service Communication Infrastructure

Figure 15. Integration with PHP Application

7.7.1. Application Logic

In this scenario departmental application will be based on three tier architecture. It will be using pearl functions to implement business logic. Application logic and user interface will be clearly separated using a well defined interface. Pearl functions may be packaged as static or dynamically linked library.

7.7.2. User Interface

Web based user interface will be implemented using PHP (pearl home pages).

7.7.3. Application DB

Technologies like MySQL, Oracle, SQL server, DB2 etc. will be used in the data tier.

7.8. Service Level Integration

In this integration approach State Portal (external application) communicates with an existing application's ready to use services, which are implemented as web services. State Portal would not be required to develop any new service or modify any part of existing application; it just invokes services provided by the departmental application.

State government department's applications, which are developed recently using web services technology and service oriented architecture would be the good candidates for adopting service oriented integration approach.

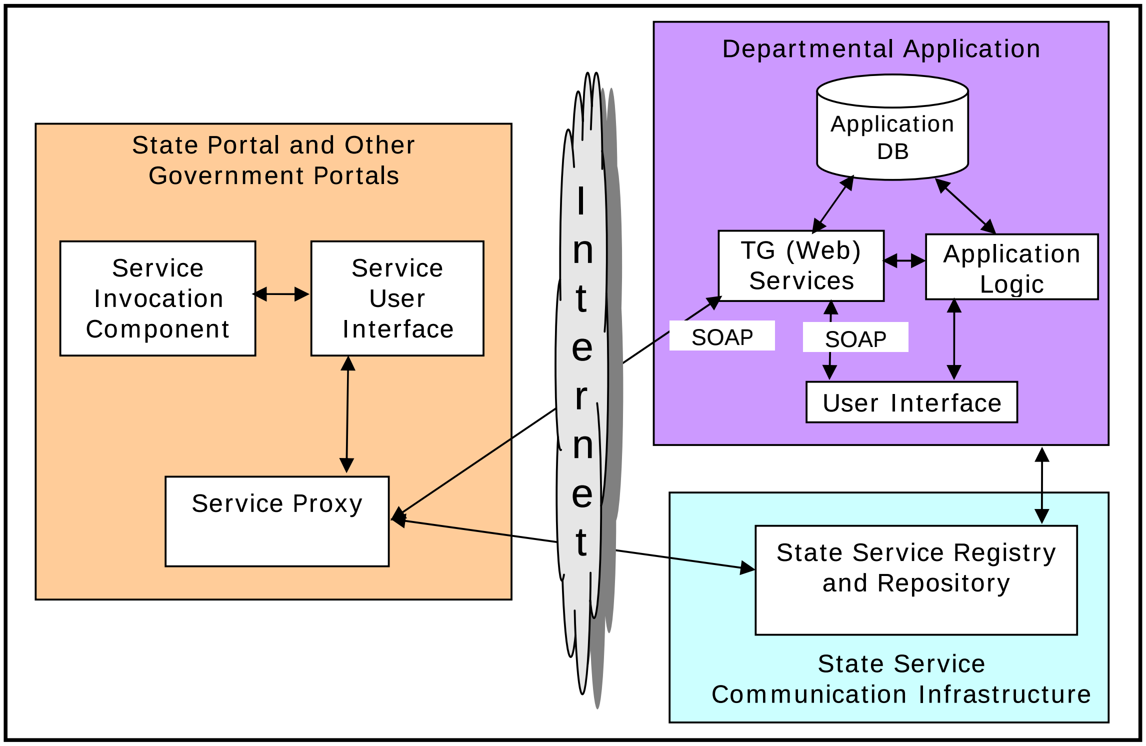
Departmental Application
Application DB
TG (Web) Services
Application Logic
SOAP
SOAP
User Interface
State Portal and Other Government Portals
Service Invocation Component
Service User Interface
Service Proxy
Internet
State Service Registry and Repository
State Service Communication Infrastructure

Figure 16. Service Level Integration

7.8.1. Application Logic

In this scenario departmental application will be based on three tier architecture. It will be using web service to expose and implement business logic. Web services may or may not use of components. Existing user interface will also be making use of same web services.

7.8.2. User Interface

In this scenario most likely application will be providing web based user interface.

7.8.3. Application DB

Technologies used for application DB will be Oracle, SQL server, DB2 etc.

7.9. Forthcoming Applications

Departmental applications which will be developed in future should be easily accessible to State Portals. State Portal should be able to invoke or use functionality or these applications or processes and procedures implemented by these applications. For invoking the services provided by departmental application from State Portal should not require any changes to existing code or development on new code on departmental applications side.

Core principle for making new departmental applications accessible to citizens will be, exposing services provided by state departments as web-services and building applications using them. This does not mean everything should be developed as a web-service. Proper due diligence should be done and only citizens or businesses related functionality should be developed as web-services. Secondly volume of service requests and usefulness or value of services to citizens or businesses should also be considered.

Adherence to following guidelines while developing new departmental applications would make them easily accessible to State Portals.

a. Develop applications based on multi layer architecture, separating presentation, business logic and data from each other using a well defined interface.
b. Provide browser based user interface.
c. Use open standards where ever possible. For example use JDBC or OBDC for accessing data from business logic layer.
d. Undertake service identification exercise as part of architecture definition phase to identify high value transactional government services. Realize services using the right combination of atomic and composite service.
e. Implement web-services so as to enable development of composite applications.
f. Implement citizen or businesses related functionality as web services.
g. Keep loose coupling between application and transactional government services.

7.10. Levels of Service Reuse

In a country like India with great degree of diversity in terms of culture and geography, state and central governments, enterprises etc. transactional government services would be expected to vary from state to state in terms of nature of services, quality of service, underlying processes, procedures etc. I this type of situation State Portals would reuse services at different level based on specific needs.

7.10.1. Level-1 Application Form Reuse

Every service by government would have some sort of associated application/request forms. All forms would be published on State Portals. When creating new forms already published forms by other states would be referred to use them as template, off course changing state specific content such as state logo.

7.10.2. Level-2 Specifications Reuse

Specifications of transactional government services would be published on service repositories. While preparing specifications of new transactional government services with similar service, functionality or purpose, existing specifications would be referred to use them in as-is form or with modifications. It would be desirable to have common version of service specifications.

7.10.3. Level-3 Code Reuse

Source code developed to implement transactional government services would be managed with version control. While developing transactional government services with similar service specifications, already available source code would be referred to reuse in as-is form or with modifications. It would be desirable to have common version of source code.

7.10.4. Level-4 Service Instance Reuse

At higher maturity levels, services would be customizable and configurable such that new instance of a service can be reused in another state, another department or another application.

7.10.5. Level-5 Runtime Reuse

The highest level of reuse where same instance of service is used by all its consumers. This would be most desirable scenario.